

УДК 004.4**В.В. Яцишин, канд.техн.наук, доц, Р.Б. Ладика, канд.фіз.-мат. наук, доцент**¹Тернопільський національний технічний університет імені Івана Пулюя, Україна²Тернопільський державний медичний університет імені І.Я. Горбачевського, Україна**СУЧАСНІ ТЕНДЕНЦІЇ РОЗВИТКУ CLOUD COMPUTING**

V.V. Yatsyshyn, PhD, Assoc. Prof., R.B. Ladyka, PhD, Assoc. Prof.

MODERN TRENDS IN CLOUD COMPUTING

У сучасних умовах розвитку ІТ-індустрії, інженерія програмного забезпечення потребує залучення нових підходів до проектування ПЗ, модернізації робочого циклу розробки проектів, скорочення затрат на випуск продукту або його частин (у випадку аутсорсингу), здешевлення вартості виконання робіт на окремих етапах і стадіях життєвого циклу. Окрім цього, важливою є адаптація організаційних та додаткових процесів життєвого циклу до сучасних викликів у галузі ІТ-індустрії. Зокрема, це пов'язано з інтенсивним розвитком IoT технологій, cloud сервісів, хмарних обчислень, ростом даних, які необхідно швидко опрацьовувати і які, зазвичай, є неоднорідними та розподіленими. Тому, на сьогодні актуальними задачами як в науковому, так і в бізнес контекстах є розробка і впровадження технологій cloud computing і їх реалізацій, які б дали змогу оптимізувати розробку програмних систем на усіх рівнях життєвого циклу з врахуванням особливостей існуючих у компаніях моделей зрілості процесів.

На даний час cloud сервіси надають широкі можливості щодо зберігання та опрацювання великих об'ємів інформації, забезпечення обміну даних між різними джерелами інформації, взаємодії різних сервісів, в тому числі інтелектуальних. Однак, для ефективного використання «хмарних технологій» в контексті реалізації бізнес процесів та витрат за використання ресурсів необхідно залучати фахівців з високим рівнем кваліфікації у сфері cloud computing. На даний час таких фахівців доволі невелика кількість, що пов'язано з низкою факторів. Зокрема, це стосується відсутності або недосконалістю документації щодо використання тих чи інших cloud сервісів, фахівці з програмування не мають достатнього досвіду адміністрування та розгортання сервісів, а у фахівців з системного адміністрування – недостатня база в інженерії програмного забезпечення. Для зниження порогу входу як програмістів, так і системних адміністраторів необхідно створювати проміжні шари між «хмарою» з відповідними програмними сервісами та програмних забезпеченням, яке проектується. Тому актуальність створення інструментальних засобів, а інколи і цілих платформ розробки програмного забезпечення з підтримкою cloud сервісів є актуальною задачею.

Проведемо аналіз платформи Onlizer, яка проектується за участі авторів, як системи взаємодії з cloud сервісами з однієї сторони і платформи розробки web-орієнтованого програмного забезпечення з іншої.

Onlizer – online платформа розробки та супроводу програмного забезпечення, побудована на мікросервісній архітектурі з орієнтацією на використання мультиагентів та cloud сервісів, що в комплексі представляє собою PaaS. Надає можливості взаємодії із сховищами даних, зокрема реляційними та документоорієнтованими структурами, апаратним забезпеченням з доступом до Internet (підтримки IoT), володіє набором візуальних інструментів для побудови back-end логіки web-орієнтованого програмного забезпечення. Onlizer містить market place конекторів, у якому, окрім власних конекторів, можна публікувати конектори сторонніх розробників. Є хорошим інструментом для інтеграції однорідного та неоднорідного програмного забезпечення, а також міграції «наземного» ПЗ в «хмару».

Найбільш важливим і трудомістким процесом життєвого циклу є процес збору та аналізу вимог до програмного забезпечення, оскільки від результатів його виконання залежать якість майбутньої програмної системи, терміни і вартість реалізації. Оскільки, процес збору та аналізу вимог, завжди супроводжується внесенням змін у специфікацію вимог, то важливим є швидкість і простота внесення змін в архітектуру ПЗ. При використанні Onlizer, швидкість внесення змін в архітектуру ПЗ, у порівнянні з класичними середовищами розробки, які вимагають написання програмного коду, в середньому зростає на 60%. Швидкість і простота внесення змін в архітектуру обумовлено застосуванням візуальних компонентів, у яких вказуються параметри та налаштування, необхідні для реалізації тієї чи іншої вимоги

Архітектура програмного забезпечення є важливим аспектом розробки програмного забезпечення. До архітектури висувають ряд вимог, найбільш вагомими з яких є: масштабованість; простота; зрозумілість; здатність до трансформації.

Окрім цього, при проектуванні архітектури необхідно забезпечити високу зв'язність всередині окремо взятого модуля та мінімальне зчеплення між модулями.

Onlizer надає засоби масштабування архітектури і практично миттєво публікує їх в нову версію розроблюваного ПЗ. Оскільки, кожен компонент системи, що проектується на основі Onlizer, представляє собою сервіс, який взаємодіє з іншим через API, то реалізація масштабування архітектури в середньому проводиться на 40% швидше, ніж при застосуванні класичних підходів з написання коду.

Здатність до трансформації архітектури, як і у випадку її масштабованості, обумовлено використанням компонентів як сервісів. Зв'язність всередині програмних модулів проектованої системи забезпечується уже на рівні самої платформи, оскільки компоненти є повторно використовуваними, а це означає, що вони пройшли верифікацію практикою і засвідчили свою надійність. Зчеплення між модулями проектованої системи повинні забезпечувати самі розробники, виходячи із реальних задач, які перед ними постають.

Процес кодування архітектури програмних систем на основі платформи Onlizer за критеріями продуктивності та ефективності зростає на 65%, оскільки написання backend логіки не вимагає написання значної кількості стрічок коду, а особливості опису предметної області та процесів, які в ній протікають займають максимум 35%. Frontend логіка програмних систем реалізується шляхом написання власних шаблонів представлення і відображення інформації, а також забезпеченням точки під'єднання до backend логіки.

Процес тестування програмних систем на основі Onlizer організований у вигляді набору інструментів для перегляду та аналізу тестових наборів даних в «near real time» як при тестуванні окремо взятих сервісів, так і об'єднаних у компоненти більш високого рівня абстракції. На кожному сервісі та зв'язку між сервісами в автоматичному режимі можна проводити аналіз параметрів і значень даних, а також інспектувати продуктивність роботи сервісів. Окрім цього, як і при класичному підході до тестування програмних систем, за допомогою Onlizer можна будувати сценарії перевірки правильності роботи компонентів на рівні unit тестування, інтеграційного та системного тестування. Час і витрати для проведення процесу тестування та інспекції при використанні Onlizer на 30% нижчі відносно класичного підходу, при цьому повнота результатів є дещо більшою, оскільки існує можливість використання службової інформації, зокрема при роботі з різними СКБД та сховищами даних.